

P&S Mobile Genomics

Lecture 7: SneakySnake

Dr. Mohammed Alser

 @mealser














ETH Zurich

Fall 2022

29 November 2022

Previous Lectures

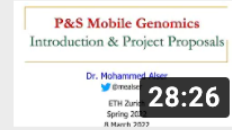

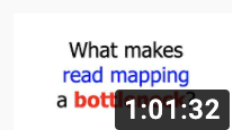
Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting
W1	8.3 Tue.	 Live	M1: P&S Mobile Genomics Course Introduction & Project Proposals  (PDF)  (PPT)
W2	15.3 Tue.	 Live	M2: Introduction to Sequencing  (PDF)  (PPT)
W3	22.3 Tue.	 Live	M3: Read Mapping  (PDF)  (PPT)
W4	29.3 Tue.	 Live	M4: GateKeeper  (PDF)  (PPT)  Video

Livestream - P&S Genome Sequencing on Mobile

Onur Mutlu Lectures - 1 / 3



- 
Mobile Genomics Course - Meeting 1: Course Introduction ...
 Onur Mutlu Lectures
- 
Mobile Genomics Course - Meeting 2: Introduction to...
 Onur Mutlu Lectures
- 
Mobile Genomics Course - Meeting 3: Read Mapping (Sprin...
 Onur Mutlu Lectures

https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php

[?id=genome_seq_mobile](#)

Goal: Minimizing Alignment Time

Sequence Alignment is expensive

Our goal is to accelerate read mapping
by reducing the need for
dynamic programming algorithms

Key Idea

Genomic Strings

```
graph TD; A[Genomic Strings] --> B[Dissimilar Strings]; A --> C[Similar Strings]; B --> D[Ignore them if the number of differences exceeds a threshold.]; C --> E[Find number, location, and type of differences?];
```

EXPENSIVE!

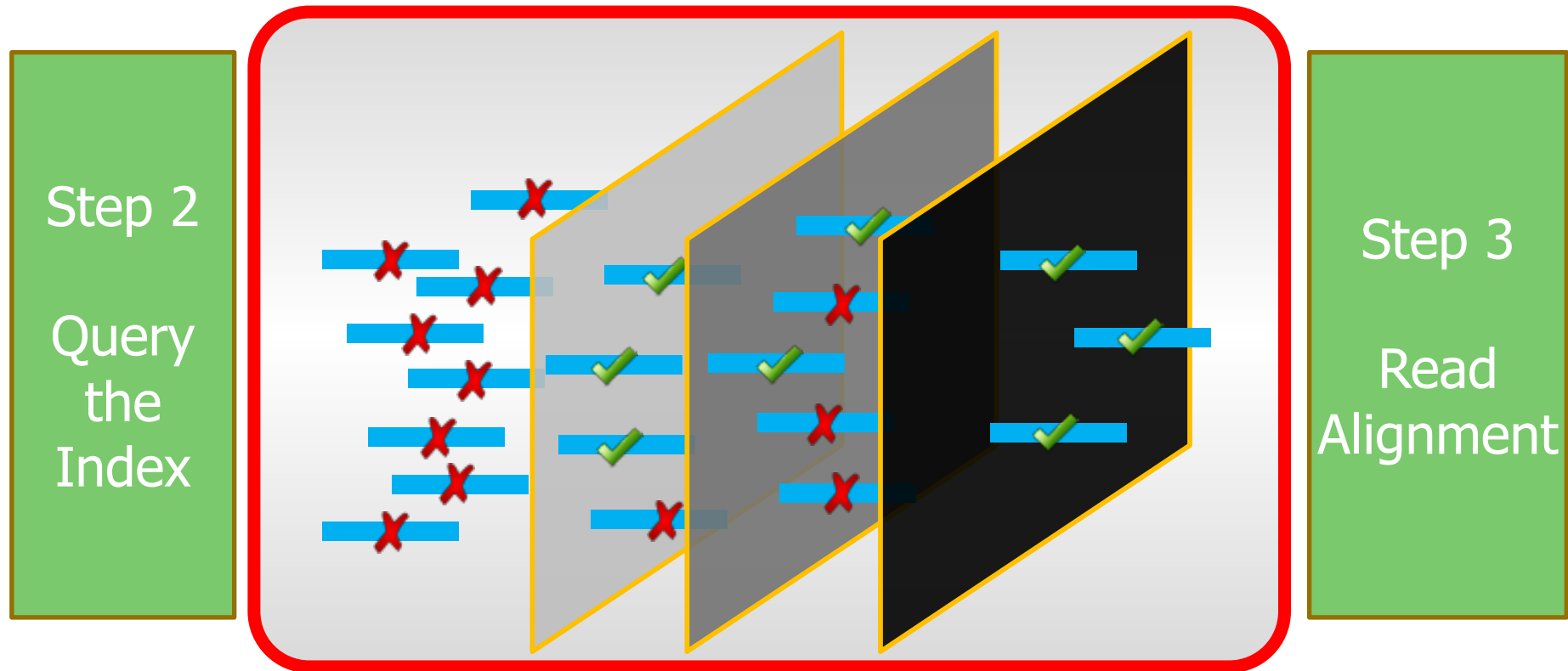
Dissimilar
Strings

Ignore them if the number
of differences exceeds a
threshold.

Similar
Strings

Find number, location, and
type of differences?

Ideal Filtering Algorithm

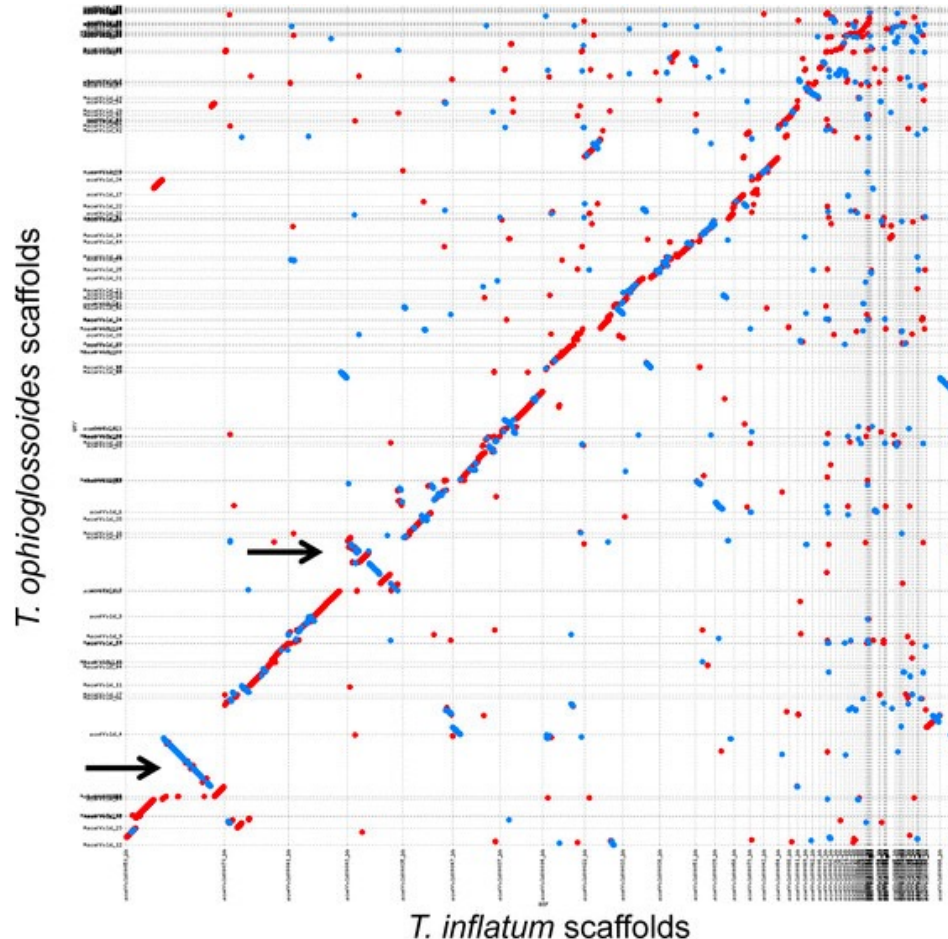


1. **Filter out** most of incorrect mappings.
2. **Preserve** all correct mappings.
3. Do it **quickly**.

SneakySnake

■ Key observation:

- Correct alignment is a sequence of non-overlapping long matches.



Dot plot, dot matrix
(Lipman and Pearson, 1985)

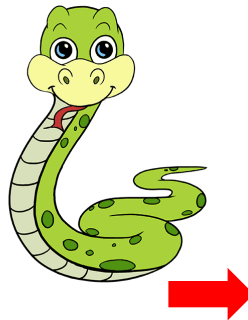
SneakySnake

- **Key observation:**

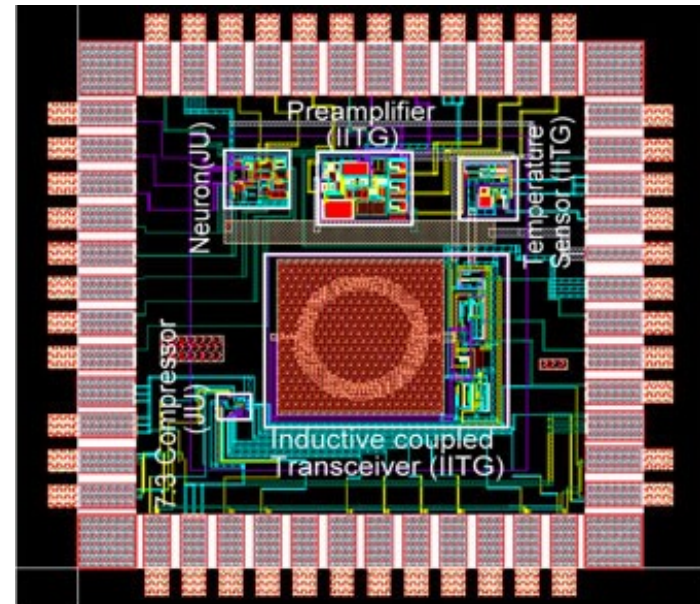
- Correct alignment is a sequence of non-overlapping long matches

- **Key idea:**

- Approximate edit distance calculation is similar to **Single Net Routing problem** in VLSI chip



VLSI chip layout



SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival

Given two genomic sequences, a reference sequence $R[1 \dots m]$ and a query sequence $Q[1 \dots m]$, and an edit distance threshold E , we calculate the entry $Z[i, j]$ of the chip maze, where $1 \leq i \leq (2E + 1)$ and $1 \leq j \leq m$, as follows:

$$E = 3$$

$$Z[i, j] = \begin{cases} 0, & \text{if } i = E + 1, Q[j] = R[j], \\ 0, & \text{if } 1 \leq i \leq E, Q[j - i] = R[j], \\ 0, & \text{if } i > E + 1, Q[j + i - E - 1] = R[j], \\ 1, & \text{otherwise} \end{cases} \quad (1)$$

column	1	2	3	4	5	6	7	8	9	10	11	12
<i>3rd Upper Diagonal</i>	1	1	1	0	1	1	0	0	0	1	1	1
<i>2nd Upper Diagonal</i>	1	1	1	0	1	1	1	1	1	1	0	1
<i>1st Upper Diagonal</i>	1	0	1	1	1	0	0	0	0	1	0	1
<i>Main Diagonal</i>	0	0	0	0	1	1	1	1	1	1	1	1
<i>1st Lower Diagonal</i>	0	1	1	1	1	0	0	1	1	1	0	1
<i>2nd Lower Diagonal</i>	1	0	1	0	1	1	1	1	0	1	1	1
<i>3rd Lower Diagonal</i>	0	1	1	1	1	1	1	1	1	1	1	1

SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival

$$E = 3$$

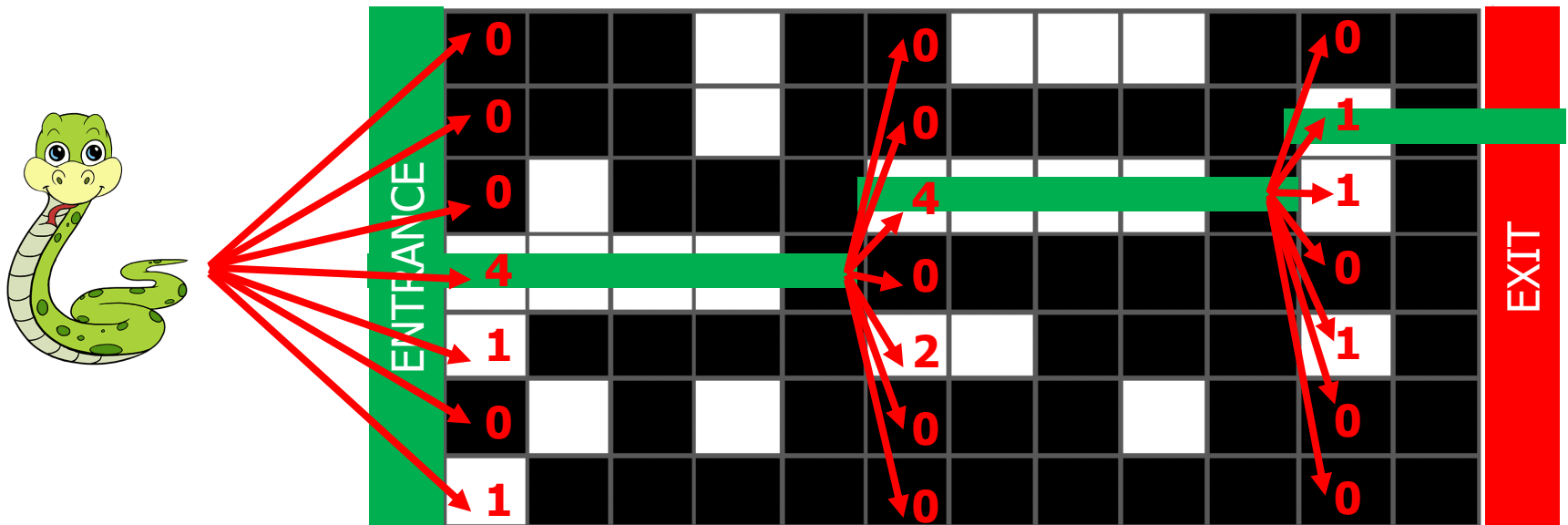
	column	1	2	3	4	5	6	7	8	9	10	11	12	
<i>3rd Upper Diagonal</i>	ENTRANCE	■	■	■	□	■	■	□	□	□	■	■	■	EXIT
<i>2nd Upper Diagonal</i>		■	■	■	□	■	■	■	■	■	■	□	■	
<i>1st Upper Diagonal</i>		■	□	■	■	■	□	□	□	□	■	□	■	
<i>Main Diagonal</i>		□	□	□	□	■	■	■	■	■	■	■	■	
<i>1st Lower Diagonal</i>		□	■	■	■	■	□	□	■	■	■	□	■	
<i>2nd Lower Diagonal</i>		■	□	■	□	■	■	■	■	□	■	■	■	
<i>3rd Lower Diagonal</i>		□	■	■	■	■	■	■	■	■	■	■	■	

SneakySnake Walkthrough

Building Neighborhood Map

Finding the Optimal Routing Path

Examining the Snake Survival



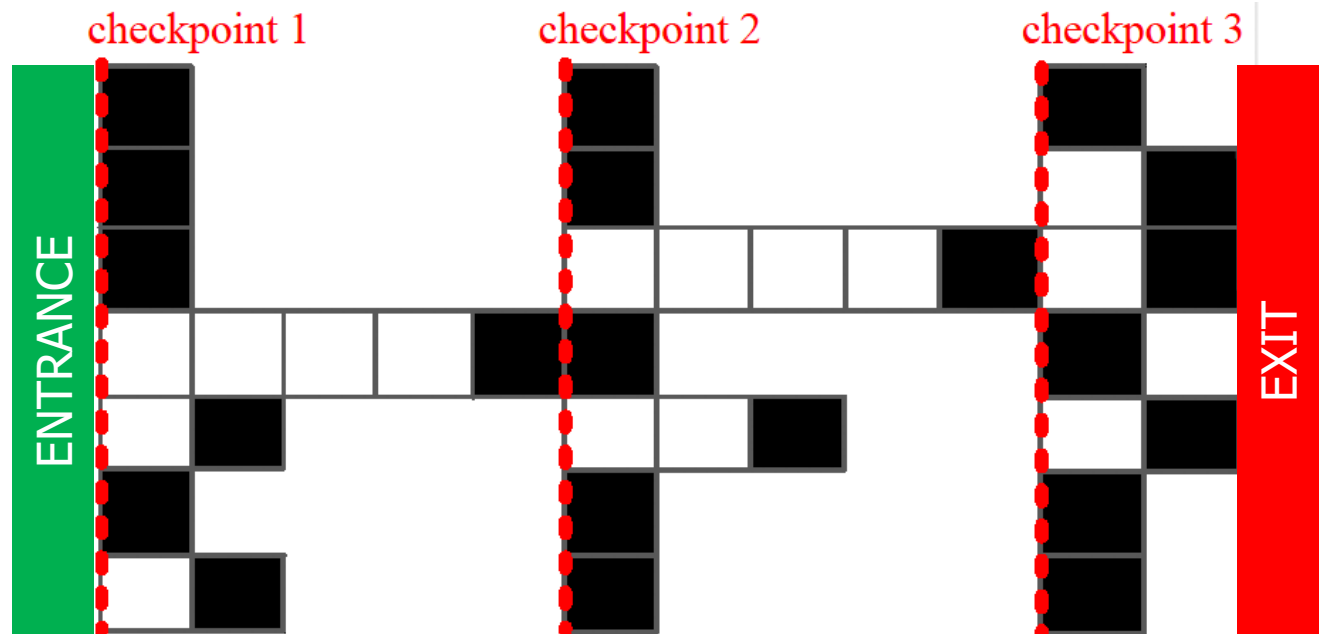
SneakySnake Walkthrough

Building Neighborhood Map

Finding the Routing Travel Path

Examining the Snake Survival

This is what you actually need to **build**
and it can be done **on-the-fly!**

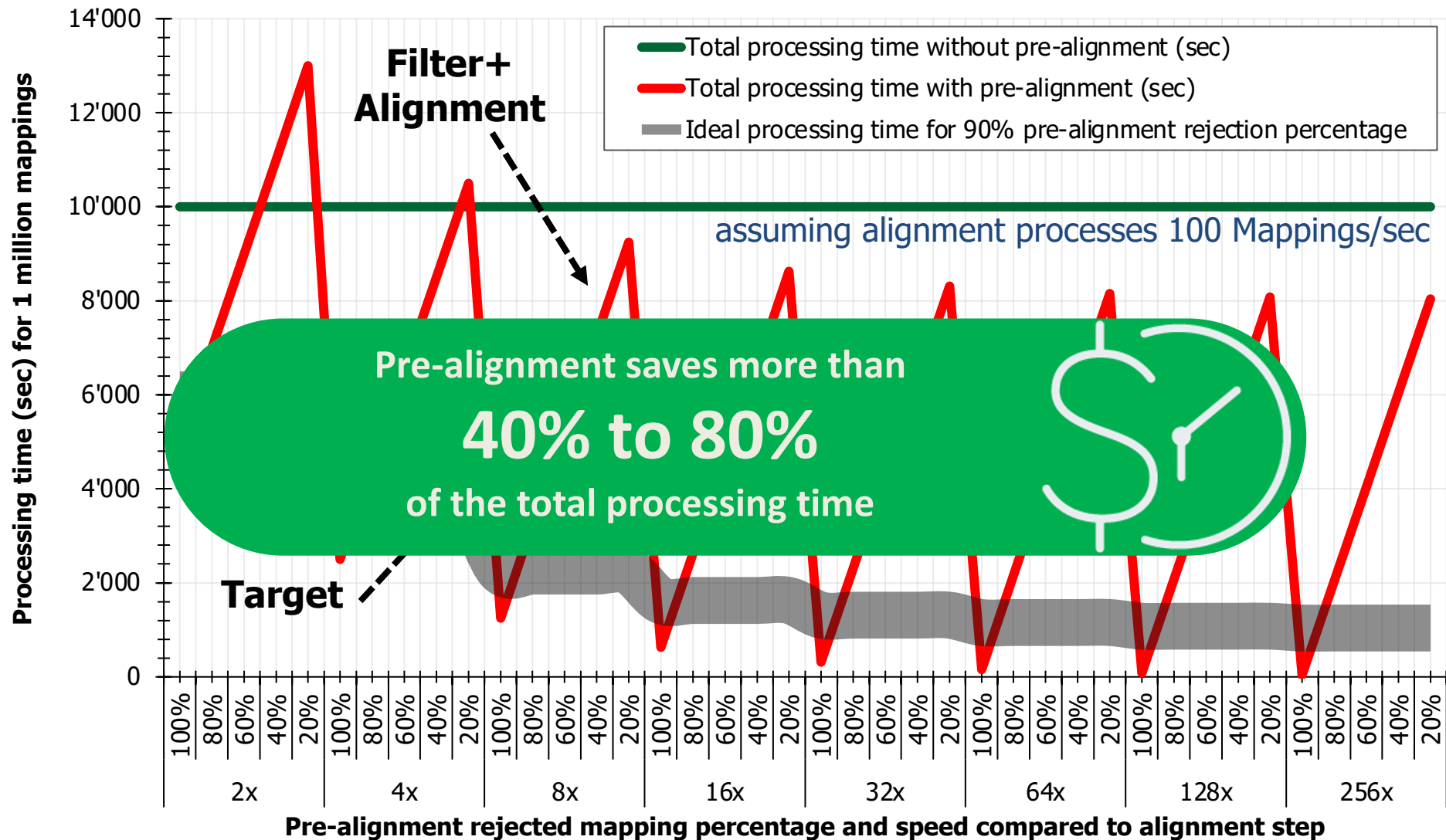


FPGA Resource Analysis

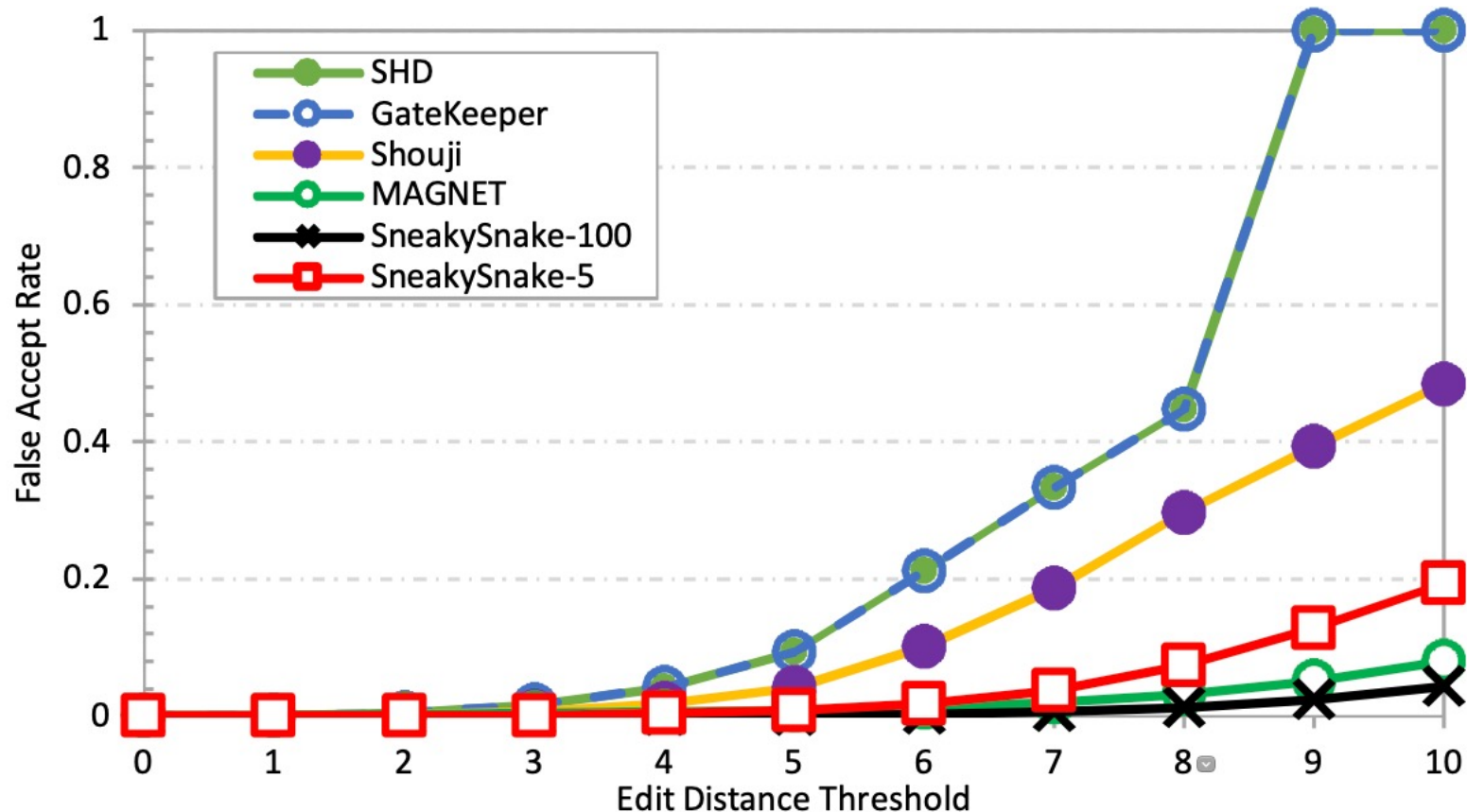
- FPGA resource usage for a single filtering unit of GateKeeper, Shouji, and Snake-on-Chip for a sequence length of 100 and under different edit distance thresholds (E).

	E (bp)	Slice LUT	Slice Register	No. of Filtering Units
GateKeeper	2	0.39%	0.01%	16
	5	0.71%	0.01%	16
Shouji	2	0.69%	0.08%	16
	5	1.72%	0.16%	16
Snake-on-Chip	2	0.68%	0.16%	16
	5	1.42%	0.34%	16

The Effect of Pre-Alignment (Theoretically)

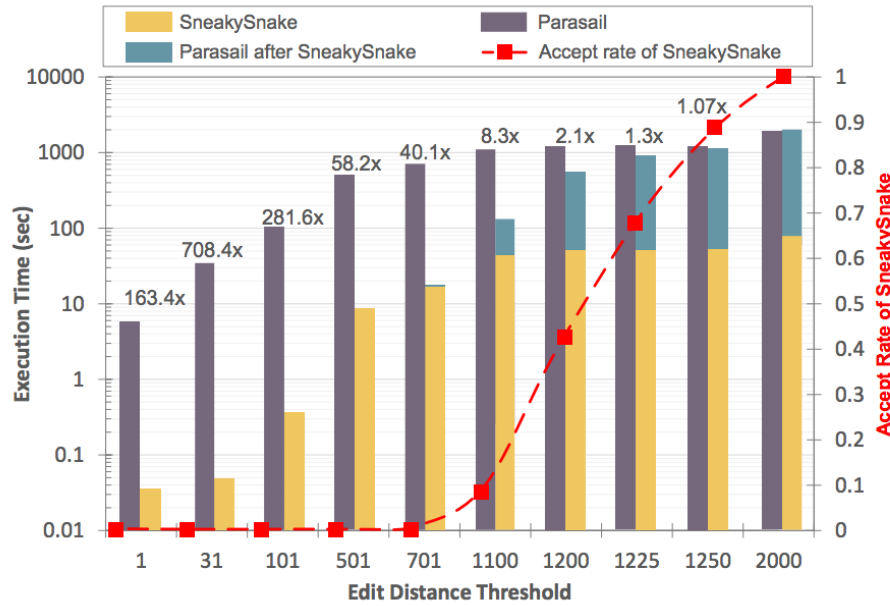


Filtering Accuracy



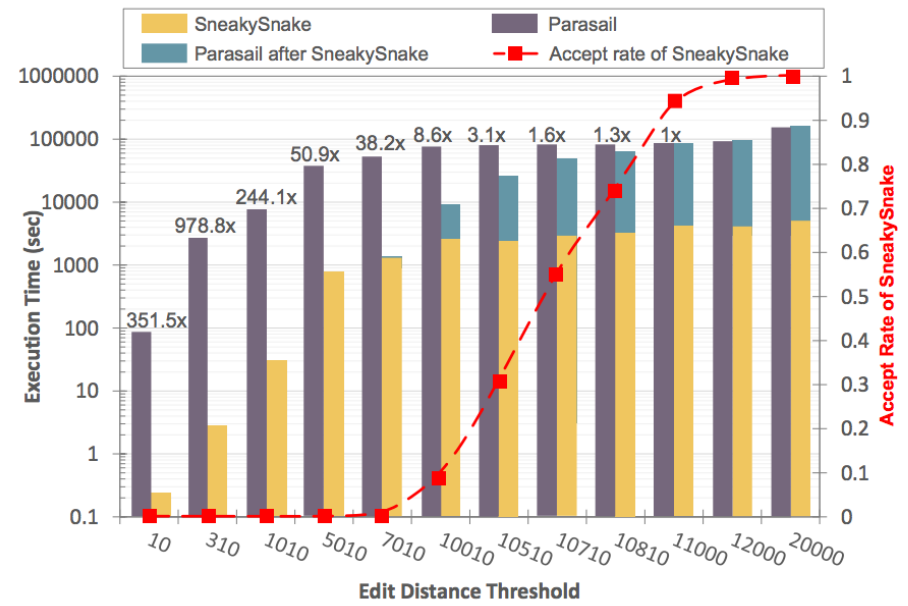
Long Read Mapping (SneakySnake vs Parasail)

10K bp reads



(a)

100K bp reads

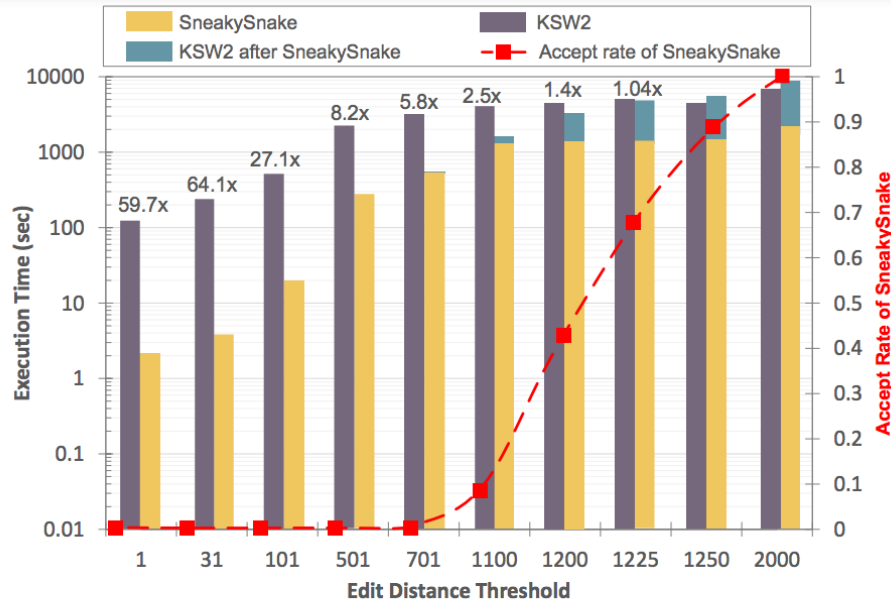


(b)

Fig. 10: The execution time of SneakySnake, Parasail, and SneakySnake integrated with Parasail using long sequences, (a) 10Kbp and (b) 100Kbp, and 40 CPU threads. The left y-axes of (a) and (b) are on a logarithmic scale. For each edit distance threshold value, we provide in the right y-axes of (a) and (b) the rate of accepted pairs (out of 100,000 pairs for 10Kbp and out of 74,687 pairs for 100Kbp) by SneakySnake that are passed to Parasail. We present the end-to-end speedup values obtained by integrating SneakySnake with Parasail.

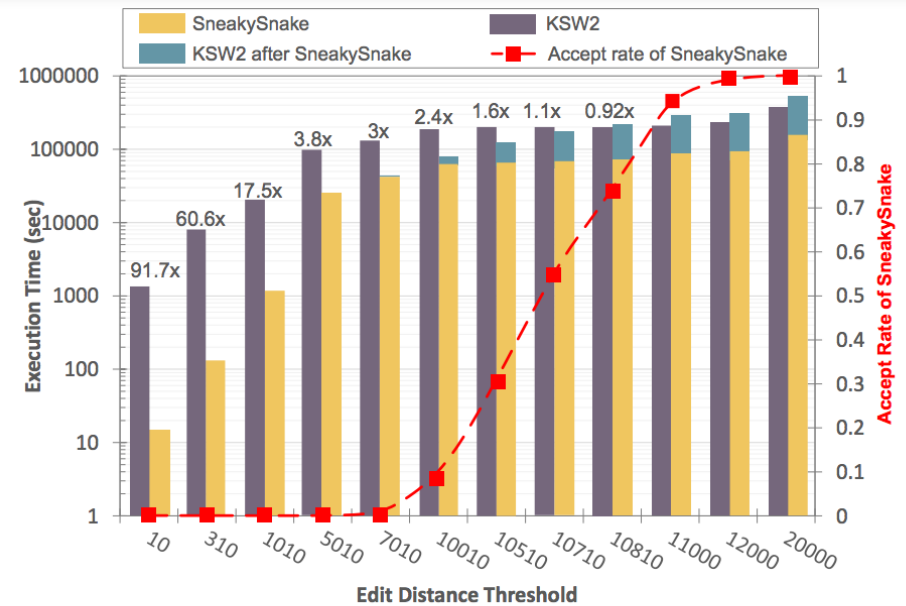
Long Read Mapping (SneakySnake vs KSW2)

10K bp reads



(a)

100K bp reads



(b)

Fig. 11: The execution time of SneakySnake, KSW2, and SneakySnake integrated with KSW2 using long sequences, (a) 10Kbp and (b) 100Kbp, and a single CPU thread. The left y-axes of (a) and (b) are on a logarithmic scale. For each edit distance threshold value, we provide in the right y-axes of (a) and (b) the rate of accepted pairs (out of 100,000 pairs for 10Kbp and out of 74,687 pairs for 100Kbp) by SneakySnake that are passed to KSW2. We present the end-to-end speedup values obtained by integrating SneakySnake with KSW2.

Key Results of SneakySnake

- ❑ SneakySnake is up to **four orders of magnitude more accurate** than **Shouji** (Bioinformatics'19) and **GateKeeper** (Bioinformatics'17)
- ❑ Using short reads, SneakySnake **accelerates Edlib** (Bioinformatics'17) and **Parasail** (BMC Bioinformatics'16) by
 - up to **37.7× and 43.9×** (>12× on average), on CPUs
 - up to **413× and 689×** (>400× on average) with **FPGA/GPU acceleration**
- ❑ Using long reads, SneakySnake **accelerates Parasail** and **KSW2** by **140.1× and 17.1×** on average, respectively, on CPUs

Alleviating Data Movement Bottlenecks

Read Mapping & Filtering in Memory

We need to design
mapping & filtering algorithms
that fit processing-in-memory

Near-memory Pre-alignment Filtering

Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gomez-Luna, Henk Corporaal, Onur Mutlu,

"FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications"

IEEE Micro, 2021.

[[Source Code](#)]



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2021.04](#)

IEEE Micro

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)

Authors

[Gagandeep Singh](#), ETH Zürich, Zürich, Switzerland

[Mohammed Alser](#), ETH Zürich, Zürich, Switzerland

[Damla Senol Cali](#), Carnegie Mellon University, Pittsburgh, PA, USA

[Dionysios Diamantopoulos](#), Zürich Lab, IBM Research Europe, Rüschlikon, Switzerland

[Juan Gomez-Luna](#), ETH Zürich, Zürich, Switzerland

[Henk Corporaal](#), Eindhoven University of Technology, Eindhoven, The Netherlands

[Onur Mutlu](#), ETH Zürich, Zürich, Switzerland

◀	▶
Previous	Next
☰	Table of Contents
📖	Past Issues

Near-memory SneakySnake

- Problem: Read Mapping is heavily bottlenecked by data movement from main memory
- Solution: Perform read mapping near where data resides (i.e., near-memory)
- We carefully redesigned the accelerator logic of SneakySnake to exploit near-memory computation capability on modern FPGA boards with high-bandwidth memory

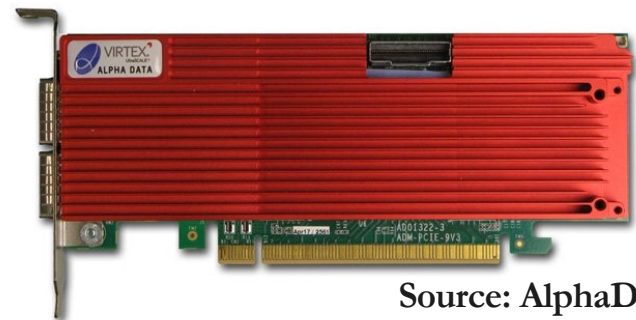
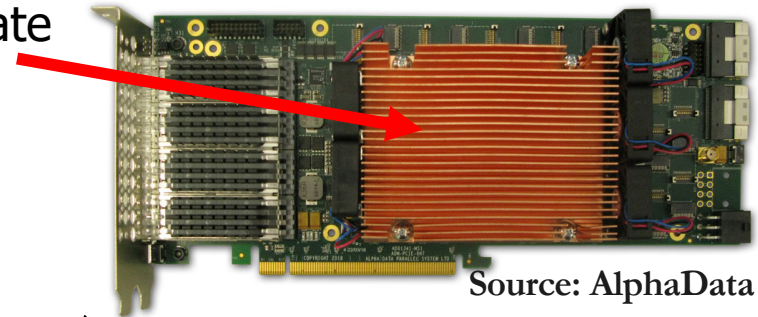
Heterogeneous System: CPU+FPGA

We evaluate two POWER9+FPGA systems:

1. **HBM-based AD9H7 board:** Xilinx Virtex Ultrascale+™ XCVU37P-2
2. **DDR4-based AD9V3 board:** Xilinx Virtex Ultrascale+™ XCVU3P-2

FPGA + HBM on the same package substrate

HBM-based AD9H7 board

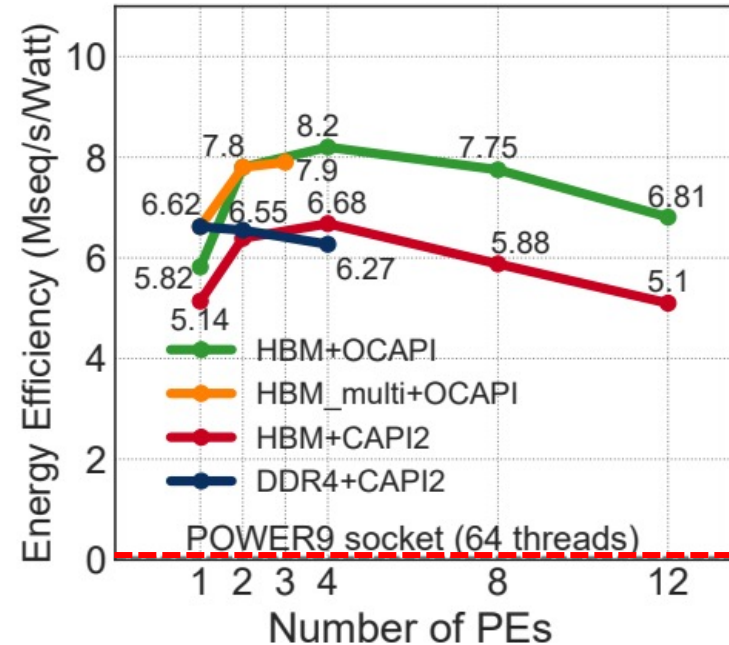
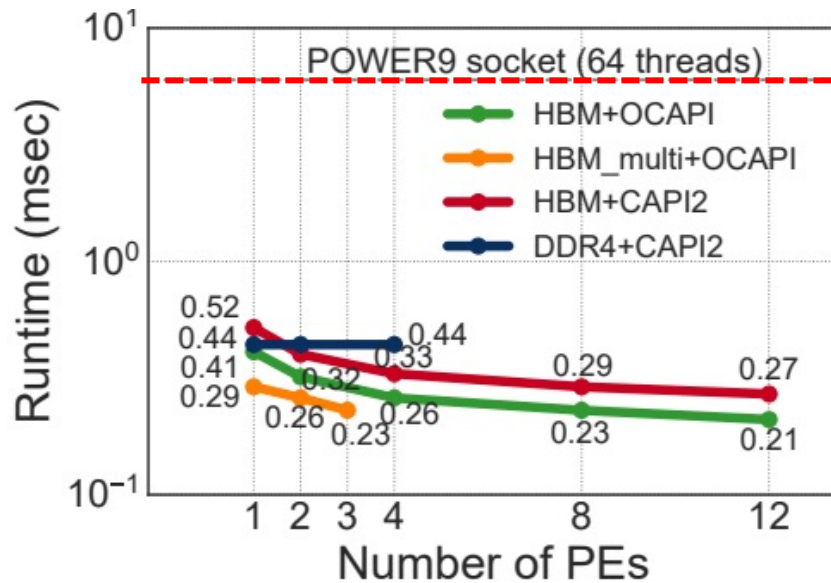


DDR4-based AD9V3 board



POWER9 AC922

Key Results of Near-memory SneakySnake



Near-memory pre-alignment filtering improves **performance** and **energy efficiency** by 27.4× and 133×, respectively, over a 16-core (64 hardware threads) IBM POWER9 CPU

Near-memory Pre-alignment Filtering

Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gomez-Luna, Henk Corporaal, Onur Mutlu,

"FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications"

IEEE Micro, 2021.

[[Source Code](#)]



[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2021.04](#)

IEEE Micro

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](#)

Authors

[Gagandeep Singh](#), ETH Zürich, Zürich, Switzerland

[Mohammed Alser](#), ETH Zürich, Zürich, Switzerland

[Damla Senol Cali](#), Carnegie Mellon University, Pittsburgh, PA, USA

[Dionysios Diamantopoulos](#), Zürich Lab, IBM Research Europe, Rüschlikon, Switzerland

[Juan Gomez-Luna](#), ETH Zürich, Zürich, Switzerland

[Henk Corporaal](#), Eindhoven University of Technology, Eindhoven, The Netherlands

[Onur Mutlu](#), ETH Zürich, Zürich, Switzerland

◀	▶
Previous	Next
☰	Table of Contents
📖	Past Issues

More on SneakySnake [Bioinformatics 2020]

Mohammed Alser, Taha Shahroodi, Juan-Gomez Luna, Can Alkan, and Onur Mutlu,
**"SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment
Filter for CPUs, GPUs, and FPGAs"**

Bioinformatics, 2020.

[[Source Code](#)]

[[Online link at Bioinformatics Journal](#)]

Bioinformatics



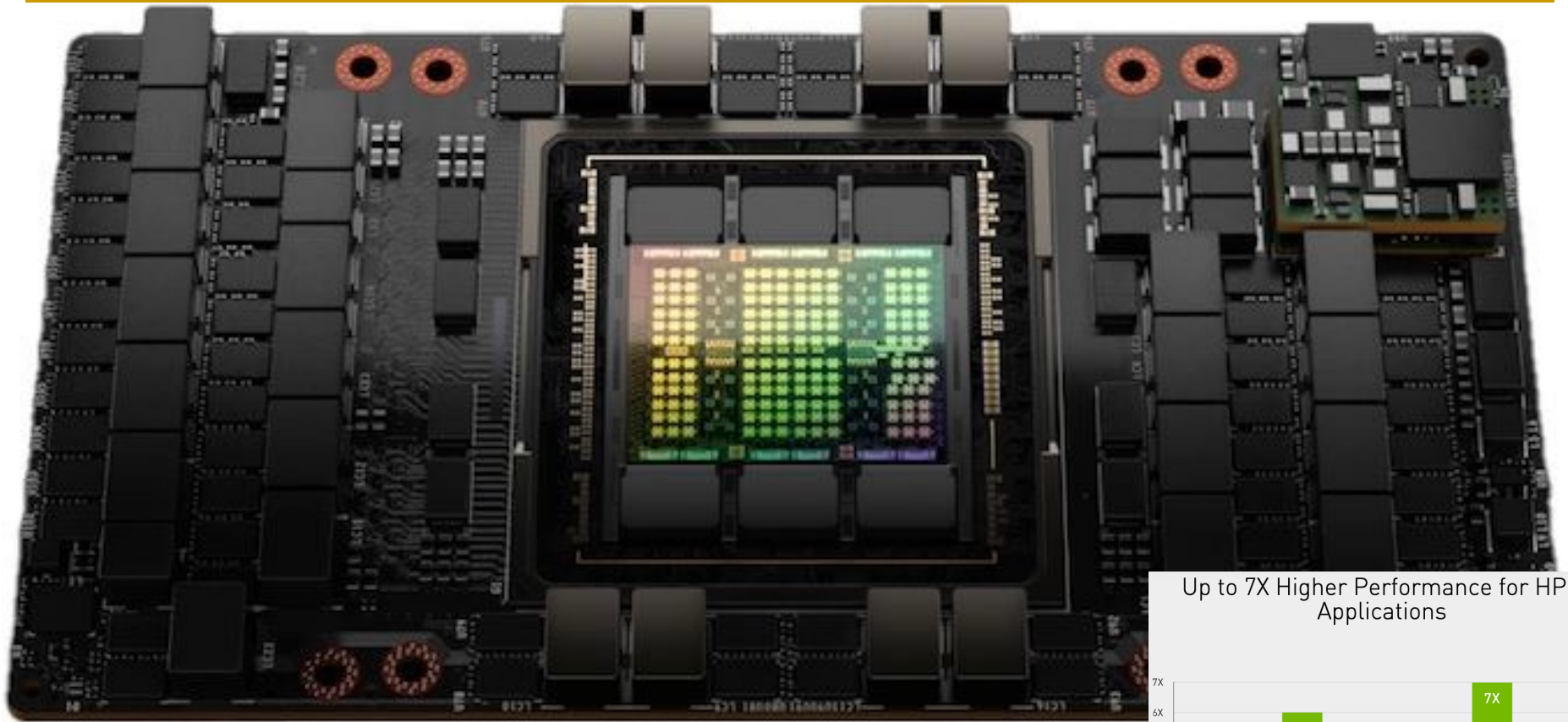
SneakySnake: a fast and accurate universal genome pre-alignment filter for CPUs, GPUs and FPGAs

Mohammed Alser ✉, Taha Shahroodi, Juan Gómez-Luna, Can Alkan ✉, Onur Mutlu ✉

Bioinformatics, btaa1015, <https://doi.org/10.1093/bioinformatics/btaa1015>

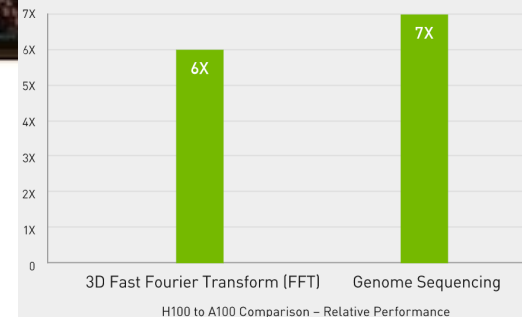
Published: 26 December 2020 **Article history** ▼

NVIDIA H100 (2022)



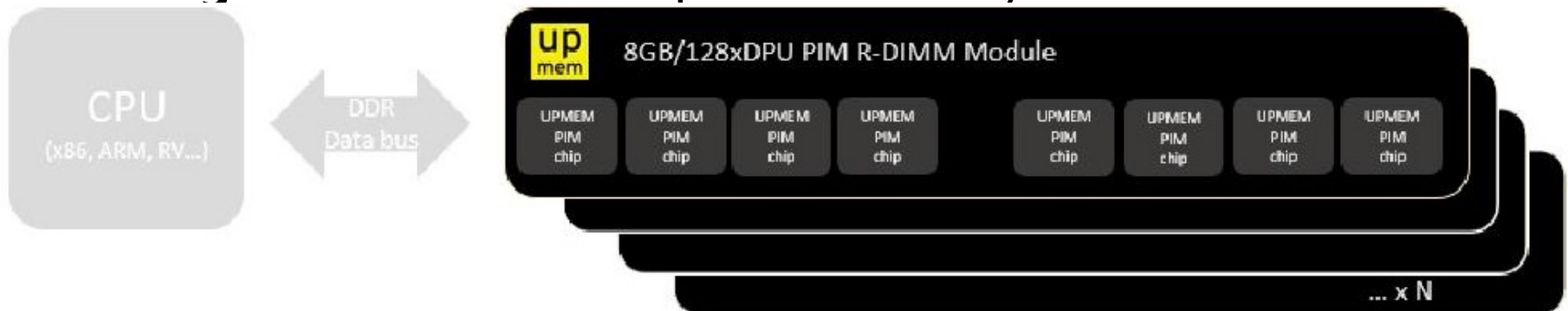
NVIDIA is claiming a **7x improvement** in dynamic programming algorithm (**DPX instructions**) performance on a single H100 versus naïve execution on an A100.

Up to 7X Higher Performance for HPC Applications



UPMEM Processing-in-DRAM Engine (2019)

- **Processing in DRAM Engine**
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.
- Replaces **standard DIMMs**
 - DDR4 R-DIMM modules
 - 8GB+128 DPUs (16 PIM chips)
 - Standard 2x-nm DRAM process
 - **Large amounts of** compute & memory bandwidth



Key Conclusion

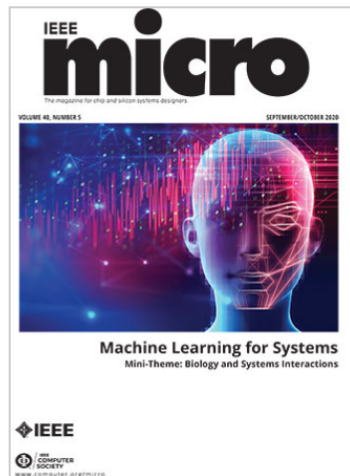
Most speedup comes from
parallelism enabled by
novel architectures and **algorithms**

Near-memory Pre-alignment Filtering

Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, Onur Mutlu

[“Accelerating Genome Analysis: A Primer on an Ongoing Journey”](#)

IEEE Micro, August 2020.



◀	▶
Previous	Next
☰	Table of Contents
📄	Past Issues

[Home](#) / [Magazines](#) / [IEEE Micro](#) / [2020.05](#)

IEEE Micro

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Sept.-Oct. 2020, pp. 65-75, vol. 40

DOI Bookmark: [10.1109/MM.2020.3013728](#)

Authors

[Mohammed Alser](#), ETH Zürich

[Zulal Bingol](#), Bilkent University

[Damla Senol Cali](#), Carnegie Mellon University

[Jeremie Kim](#), ETH Zurich and Carnegie Mellon University

[Saugata Ghose](#), University of Illinois at Urbana-Champaign and Carnegie Mellon University

[Can Alkan](#), Bilkent University

[Onur Mutlu](#), ETH Zurich, Carnegie Mellon University, and Bilkent University

Read Mapping in 111 pages!

In-depth analysis of 107 read mappers (1988-2020)

Mohammed Alser, Jeremy Rotman, Dhrithi Deshpande, Kodi Taraszka, Huwenbo Shi, Pelin Icer Baykal, Harry Taegyun Yang, Victor Xue, Sergey Knyazev, Benjamin D. Singer, Brunilda Balliu, David Koslicki, Pavel Skums, Alex Zelikovsky, Can Alkan, Onur Mutlu, Serghei Mangul

["Technology dictates algorithms: Recent developments in read alignment"](#)

Genome Biology, 2021

[[Source code](#)]

Alser et al. *Genome Biology* (2021) 22:249
<https://doi.org/10.1186/s13059-021-02443-7>


Genome Biology

REVIEW

Open Access

Technology dictates algorithms: recent developments in read alignment



Mohammed Alser^{1,2,3†}, Jeremy Rotman^{4†}, Dhrithi Deshpande⁵, Kodi Taraszka⁴, Huwenbo Shi^{6,7}, Pelin Icer Baykal⁸, Harry Taegyun Yang^{4,9}, Victor Xue⁴, Sergey Knyazev⁸, Benjamin D. Singer^{10,11,12}, Brunilda Balliu¹³, David Koslicki^{14,15,16}, Pavel Skums⁸, Alex Zelikovsky^{8,17}, Can Alkan^{2,18}, Onur Mutlu^{1,2,3†} and Serghei Mangul^{5*†} 

Feedback From Our Community!



James Ferguson

@Psy_Fer_

This is awesome! I've got my evening reading sorted.



Stéphane Le Crom

@slecrom

Very complete article on the evolution of read alignment algorithms. [#NGS](#) [#genomics](#)



Svetlana Gorokhova

@SGorokhova

An impressive overview of read alignment methods over the last three decades



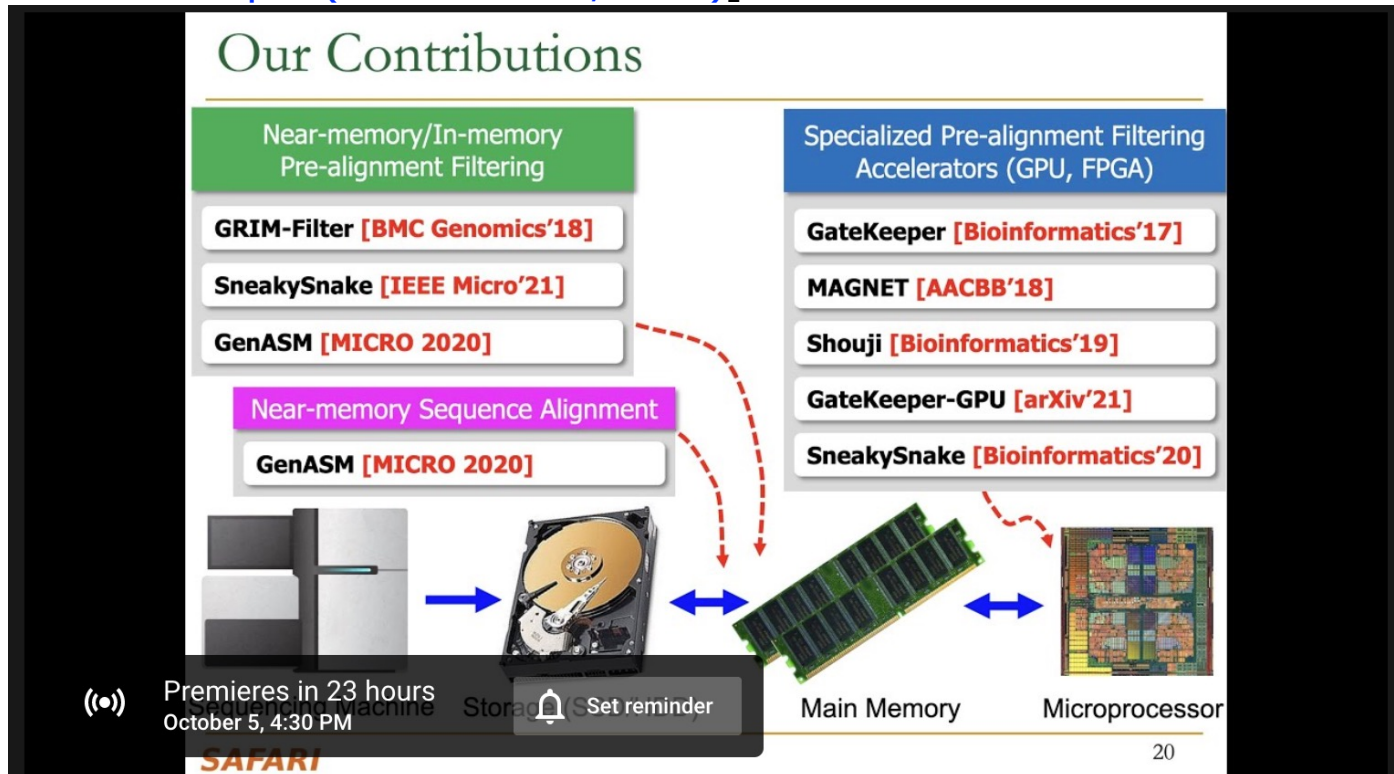
BContrerasMoreira @BrunoContrerasM · Sep 10

Replying to @mealser @GenomeBiology and 3 others

Buen hilo de repaso sobre la evolución de los algoritmos de alineamiento de secuencias a medida que ha mejorado la tecnología de secuenciación

More on Accelerating Genome Analysis ...

- Mohammed Alser,
"Accelerating Genome Analysis: A Primer on an Ongoing Journey"
Talk at [RECOMB 2021](#), Virtual, August 30, 2021.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (27 minutes)]
[[Related Invited Paper](#) (at IEEE Micro, 2020)]



More on Intelligent Genome Analysis ...

- Mohammed Alser,
"Computer Architecture - Lecture 10: Intelligent Genome Analysis"
ETH Zurich, Computer Architecture Course, Fall2021, Lecture 10, Virtual, 29 October 2021.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#) (3 hour 2 minutes, including Q&A)]
[[Related Invited Paper \(at IEEE Micro, 2020\)](#)]



Untangling Yarn Balls & DNA Sequencing

Mohammed AL...

36:59 / 3:02:00

SAFARI

zoom

Computer Architecture - Lecture 10: Intelligent Genome Analysis (Fall 2021)

412 views • Streamed live on Oct 29, 2021

19 0 SHARE SAVE ...

More on Intelligent Genome Analysis ...

- Mohammed Alser,
"Computer Architecture - Lecture 8: Intelligent Genome Analysis"
ETH Zurich, Computer Architecture Course, Lecture 8, Virtual, 15 October 2021.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#)] (2 hour 54 minutes, including Q&A)
[[Related Invited Paper \(at IEEE Micro, 2020\)](#)]

Our Solution: GateKeeper

Alignment Filter + FPGA-based Alignment Filter = 1st

Low Speed & High Accuracy
Medium Speed, Medium Accuracy
High Speed, Low Accuracy

$\times 10^{12}$ mappings

$\times 10^3$ mappings

1 High throughput DNA sequencing (HTS) technologies

2 Read Pre-Alignment Filtering
Fast & Low False Positive Rate

3 Read Alignment
Slow & Zero False Positives

108

2:08:58 / 2:54:18 • GateKeeper

ETH ZENTRUM

Computer Architecture - Lecture 8: Intelligent Genome Analysis (ETH Zürich, Fall 2020)

More on Fast Genome Analysis ...

- Onur Mutlu,
"Accelerating Genome Analysis: A Primer on an Ongoing Journey"
Invited Lecture at [Technion](#), Virtual, 26 January 2021.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (1 hour 37 minutes, including Q&A)]
[[Related Invited Paper \(at IEEE Micro, 2020\)](#)]

The video player displays a slide titled "Insight: Shifting a String Helps Similarity Search". The slide content shows two strings of letter blocks: "I STANBUL" and "I STNBUL". Green dashed arrows indicate character alignment: 'I' to 'I', 'S' to 'S', 'T' to 'T', 'A' to 'N', 'N' to 'B', 'B' to 'U', and 'U' to 'L'. This alignment shows 7 matches and 1 mismatch (A vs N). The slide number 81 is visible in the bottom right corner of the video frame.

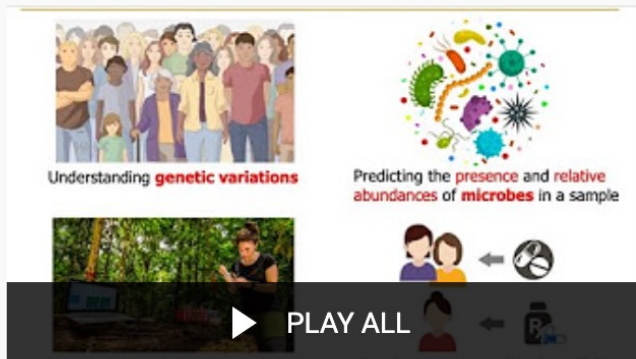
Onur Mutlu - Invited Lecture @Technion: Accelerating Genome Analysis: A Primer on an Ongoing Journey

566 views · Premiered Feb 6, 2021

Onur Mutlu Lectures
13.9K subscribers

ANALYTICS EDIT VIDEO

Two P&S Genomics Courses



Livestream - P&S Genome Sequencing on Mobile Devices (Fall 2021)




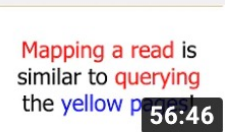

9 videos • 75 views • Updated 5 days ago



Onur Mutlu
Lectures

SUBSCRIBED



- 1  **Mobile Genomics Course - Meeting 1: Course...**
Onur Mutlu Lectures
- 2  **Intelligent Genome Analysis Course - Meeting 1: Course...**
Onur Mutlu Lectures
- 3  **Mobile Genomics Course - Meeting 2: Introduction to...**
Onur Mutlu Lectures
- 4  **Mobile Genomics Course - Meeting 3: Read Mapping...**
Onur Mutlu Lectures
- 5  **Mobile Genomics Course - Meeting 4: GateKeeper (Fall 2021)...**
Onur Mutlu Lectures

https://www.youtube.com/playlist?list=PL5Q2soXY2Zi_U2F8yrrNPD9CjcM6CFQXv

Course Materials

2021 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	5.10 Tue.	Live	M1: P&S Accelerating Genomics Course Introduction & Project Proposals (PDF) (PPT) Video	Required Materials Recommended Materials	
W2	20.10 Wed.	Live	M2: Introduction to Sequencing (PDF) (PPT)		
W3	27.10 Wed.	Live	M3: Read Mapping (PDF) (PPT)		
W4	3.11 Wed.	Live	M4: GateKeeper (PDF) (PPT)		
W5	10.11 Wed.	Live	M5: MAGNET & Shouji (PDF) (PPT)		
W6	17.11 Wed.		M6.1: SneakySnake (PDF) (PPT) Video		
			M6.2: GRIM-Filter (PDF) (PPT) Video		
W7	24.11 Wed.		M7: GenASM (PDF) (PPT) Video		

https://safari.ethz.ch/projects_and_seminars/fall2021/doku.php?id=bioinformatics

Detailed Lectures on Genome Analysis

- **Computer Architecture, Fall 2020, Lecture 3a**
 - **Introduction to Genome Sequence Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=CrRb32v7SJc&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=5>
- **Computer Architecture, Fall 2020, Lecture 8**
 - **Intelligent Genome Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=ygmQpdDTL7o&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=14>
- **Computer Architecture, Fall 2020, Lecture 9a**
 - **GenASM: Approx. String Matching Accelerator** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=XoLpzmN-Pas&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=15>
- **Accelerating Genomics Project Course, Fall 2020, Lecture 1**
 - **Accelerating Genomics** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=rgjl8ZyLsAg&list=PL5Q2soXY2Zi9E2bBVAgCqLgwiDRQDTyId>

Prior Research on Genome Analysis (1/2)

- Alser + ["SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment Filter for CPUs, GPUs, and FPGAs."](#) to appear in *Bioinformatics*, 2020.
- Senol Cali+, ["GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis"](#), *MICRO* 2020.
- Alser+, ["Technology dictates algorithms: Recent developments in read alignment"](#), to appear in *Genome Biology*, 2021.
- Kim+, ["AirLift: A Fast and Comprehensive Technique for Translating Alignments between Reference Genomes"](#), *arXiv*, 2020
- Alser+, ["Accelerating Genome Analysis: A Primer on an Ongoing Journey"](#), *IEEE Micro*, 2020.

Prior Research on Genome Analysis (2/2)

- Firtina+, "[Apollo: a sequencing-technology-independent, scalable and accurate assembly polishing algorithm](#)", *Bioinformatics*, 2019.
- Alser+, "[Shouji: a fast and efficient pre-alignment filter for sequence alignment](#)", *Bioinformatics* 2019.
- Kim+, "[GRIM-Filter: Fast Seed Location Filtering in DNA Read Mapping Using Processing-in-Memory Technologies](#)", *BMC Genomics*, 2018.
- Alser+, "[GateKeeper: A New Hardware Architecture for Accelerating Pre-Alignment in DNA Short Read Mapping](#)", *Bioinformatics*, 2017.
- Alser+, "[MAGNET: understanding and improving the accuracy of genome pre-alignment filtering](#)", *IPSI Transaction*, 2017.

P&S Mobile Genomics

Lecture 7: SneakySnake

Dr. Mohammed Alser

 @mealser

ETH Zurich

Fall 2022

29 November 2022